Claims

1. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of generating a partial procedure summary of a procedure of multithreaded software, wherein the procedure performs a plurality of actions when executed, the method comprising:

identifying a plurality of the actions as atomically modelable with respect to multithreaded execution of the procedure; and

generating the partial procedure summary of the procedure from the plurality of the actions atomically modelable with respect to multithreaded execution of the multithreaded software, wherein the partial procedure summary comprises at least one state pair, wherein the at least one state pair models execution for the procedure, and wherein the partial procedure summary of the procedure of multithreaded software is embodied on a second computer-readable medium.

2. (Cancelled)

3. (Previously Presented) The computer program product of claim 1, the method further comprising:

modeling execution of the software via the partial procedure summary.

4. (Currently Amended) The computer program product of claim 1, the method further comprising:

during modeling, comparing an indicated state invariant with a modeled state; and

responsive to determining the modeled state fails the indicated state invariant, wherein determining the modeled state fails the indicated state invariant comprises determining that a condition is false for at least one execution path, indicating that a programming flaw is present in the software.

5. (Currently Amended) The computer program product of claim 1, wherein the at least one state pair comprises an initial state and a resulting state, wherein the resulting state comprises at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure, the method further comprising:

storing an initial program counter location within the modeled procedure for the initial state;

storing a resulting program counter location within the modeled procedure for the resulting state; and

associating [[an]] the initial program counter location within the modeled procedure and [[a]] the resulting program counter location within the modeled procedure with the partial procedure summary.

6. (Currently Amended) The computer program product of claim 1, the method further comprising:

performing a reachability analysis of the software; and

consulting a procedure summary comprising the partial procedure summary when the procedure is encountered during the reachability analysis, wherein the consulting comprises determining possible execution paths within the procedure and using the procedure summary to explore possible states.

7. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying a transaction boundary within the actions.

8. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying at least one of the plurality of actions as movable later in time with respect to actions executed by other threads without affecting a resulting end state.

9. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying a sequence of actions having zero or more right movers followed by an atomic action followed by zero or more left movers.

10. (Currently Amended) The computer program product of claim 1 wherein the plurality of actions atomically modelable with respect to multithreaded execution of the software is a ~~proper~~ subset of the plurality of actions of the procedure, the subset comprising less than all of the plurality of actions of the procedure.

11. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of modeling multithreaded software, the method comprising:

analyzing actions of the multithreaded software, wherein the analyzing comprises checking the modeled multithreaded software for programming flaws; and

based on the analyzing, generating a plurality of procedure summaries for the multithreaded software, wherein the plurality of procedure summaries for the multithreaded software is embodied on a second computer-readable medium;

wherein the procedure summaries comprises a plurality of modeled ~~model~~ states of the multithreaded software for multithreaded execution of the multithreaded software.

12. (Previously Presented) The computer program product of claim 11 wherein at least one of the procedure summaries comprises at least two or more partial procedure summaries summarizing a procedure.

13. (Previously Presented) The computer program product of claim 11 wherein at least one of the procedure summaries comprises at least one partial procedure summary for a procedure, wherein the partial procedure summary summarizes less than all of the procedure.

14. (Previously Presented) The computer program product of claim 11 wherein the analyzing comprises:

identifying a series of transactions within the multithreaded software; and

modeling the transactions via partial procedure summaries.

15. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to implement a system for modeling multithreaded software, the system comprising:

a model checker operable to analyze a model of the multithreaded software via checking the model of the multithreaded software for programming flaws, the model checker comprising:

[[a]] the model of the multithreaded software, wherein the model comprises a plurality of procedure summaries modeling states of the multithreaded software during multithreaded execution of the multithreaded software, wherein the model of the multithreaded software is embodied on a second computer-readable medium.

16. (Previously Presented) The computer program product of claim 15 wherein at least one of the procedure summaries comprises a procedure summary summarizing actions deemed to have occurred one after another without interruption.

17. (Previously Presented) The computer program product of claim 15 wherein the model checker further comprises:

a reachability analyzer operable to employ the procedure summaries to generate modeled states of the software.

18. (Previously Presented) The computer program product of claim 17 wherein the system is operable to detect programming flaws via comparing an indicated state invariant with the modeled states.

19. (Currently Amended) ~~One or more~~ A computer-readable ~~media~~ medium having encoded thereon a data structure comprising:

a plurality of state pairs representing a procedure summary for multithreaded software, wherein at least one of the state pairs comprises an initial state and a resulting state indicating a state after execution of actions modeled by the procedure summary, wherein the procedure summary models multithreaded execution of the multithreaded software.

20. (Currently Amended) The computer-readable ~~media~~ medium of claim 19 wherein the state pairs comprise the following:

an indication of a first location within the procedure and an indication of a possible state for one or more variables of the multithreaded software when the procedure has reached the first location; and

an indication of a second location within the procedure and an indication of a resulting state for the one or more variables of the multithreaded software after a plurality of summarized actions of the procedure have been executed, wherein the summarized actions start at the first location and end at the second location;

wherein the plurality of summarized actions of the procedure are atomically modelable with respect to multithreaded execution of the multithreaded software.

21. (New) The computer program product of claim 1, wherein the at least one state pair comprises an initial state of the procedure and at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure.

22. (New) The computer program product of claim 1, wherein the first computer-readable medium comprises the second computer-readable medium.